Listing of Claims

Claim 1 (Currently Amended): A method of implementing atomic transactions in a system, said method comprising:

requesting in a user program ~~logic~~ a transaction identifier for an atomic transaction~~, wherein said program logic is contained in a user program designed by a programmer~~;

generating said transaction identifier in a transaction manager in response to said requesting;

specifying in said ~~program logic~~ user program a plurality of combinations ~~for execution in a sequential order~~, wherein each of said plurality of combinations contains said transaction identifier, a task procedure, and a rollback procedure, wherein said task procedure implements a part of said atomic transaction and said rollback procedure is designed to rollback said task procedure;

executing a set of ~~said~~ task procedures in ~~said~~ a sequential order according to said user program, wherein said set of task procedures are contained in said task procedures specified in said plurality of combinations;

keeping track of ~~said~~ a set of rollback procedures ~~in said transaction manager~~ corresponding to said set of task procedures, each of said set of procedures being determined based on a combination corresponding to an executed task procedure contained in said set of task procedures, said combination being contained in said plurality of combinations specified in said user program; and

executing said set of rollback procedures in a reverse order of said sequential order if said atomic transaction is to be aborted,

wherein said rollback procedure is specified as a separate procedure from said task procedure in said user program ~~wherein said rollback procedures are identified according to said keeping~~.

Claim 2 (Original): The method of claim 1, wherein said transaction identifier is unique to each of the atomic transactions.

Claim 3 (Previously Presented): The method of claim 1, wherein said keeping comprises storing data representing said rollback procedures in a stack.

Claim 4 (Original): The method of claim 3, wherein said stack is stored in a memory.

Claim 5 (Original): The method of claim 1, further comprising examining a status returned by execution of one of said task procedures and performing said aborting if said status indicates an error.

Claim 6 (Original): The method of claim 1, wherein said aborting is performed asynchronously.

Claims 7 (Currently Amended): A computer readable medium carrying one or more sequences of instructions representing a user program ~~logic~~ for execution on a system, said user program ~~logic~~ implementing an atomic transaction, wherein execution of said one or more sequences of instructions by one or more processors contained in said system causes said one or more processors to perform the actions of:

requesting an identifier in said user program from a transaction manager for said atomic transaction, wherein said transaction manager generates a unique value as said identifier;

setting a variable to equal said identifier;

specifying a plurality of combinations in said user program for execution in said system, wherein each of said plurality of combinations contains said variable ~~transaction identifier~~, a task procedure, and a rollback procedure, wherein said task procedure implements a part of said atomic transaction and said rollback procedure is designed to rollback said task procedure, wherein said variable in each of said plurality of combinations specifies said identifier generated by said transaction manager; and

aborting said atomic transaction by specifying said identifier associated with an abort procedure to cause said rollback procedures to be executed,

wherein ~~said program logic, including~~ said plurality of combinations~~;~~ and said abort procedure are contained in ~~a~~ said user program ~~designed by a programmer~~.

Claim 8 (Original): The computer readable medium of claim 7, wherein said specifying comprises including each of said plurality of combinations in a single procedure call.

Claim 9 (Original): The computer readable medium of claim 7, further comprising examining a status returned by execution of one of said task procedures and performing said aborting if said status indicates an error.

Claim 10 (Currently Amended): A computer readable medium carrying one or more sequences of instructions for supporting implementation of an atomic transaction in a system, wherein execution of said one or more sequences of instructions by one or more processors contained in said system causes said one or more processors to perform the actions of:

generating an identifier for said atomic transaction for a user program;

receiving a plurality of combinations for execution from said user program, wherein each of said plurality of combinations contains said transaction identifier, a task procedure, and a rollback procedure, wherein said task procedure implements a part of said atomic transaction and said rollback procedure is designed to rollback said task procedure;

executing a set of task procedures in a sequential order according to said user program, wherein said set of task procedures are contained in said plurality of combinations ~~said task procedures~~;

keeping track of a set of rollback procedures corresponding to said set of task procedures, each of said set of procedures being determined based on a combination corresponding to an executed task procedure contained in said set of task procedures, said combination being contained in said plurality of combinations specified in said user program; and

executing said set of rollback procedures in a reverse order of said sequential order in response to receiving an abort request,

wherein said rollback procedure is specified as a separate procedure from said task procedure in said user program.

~~wherein each of said plurality of combinations is received from a corresponding user program and wherein the roll back procedure received from a first user program is different from the roll back procedure received from a second user program.~~

Claims 11 - 12 (Canceled)

Claim 13 (Currently Amended): The computer readable medium of claim 10 ~~12~~, wherein said transaction identifier is generated to be unique for each atomic transaction.

Claim 14 (Currently Amended): The computer readable medium of claim 10 ~~12~~, wherein said ~~data is~~ set of rollback procedures are represented in the form of a stack.

Claim 15 (Original): The computer readable medium of claim 14, wherein said stack is stored in a memory.

Claim 16 (Currently Amended): A computer system comprising:

a memory storing a plurality of instructions; and

a processing unit coupled to said memory and executing said plurality of instructions to support implementation of atomic transactions in a programming environment, said processing unit being operable to:

request in a user program ~~logic~~ a transaction identifier for an atomic transaction;

generate said transaction identifier in a transaction manager in response to said requesting, wherein said transaction manager is provided external to said user program;

specify in said user program ~~logic~~ a plurality of combinations ~~for execution in a sequential order~~, wherein each of said plurality of combinations contains said transaction identifier, a task procedure, and a rollback procedure, wherein said task procedure implements a part of said atomic transaction and said rollback procedure is designed to rollback said task procedure, wherein said rollback procedure is specified as a separate procedure from said task procedure;

execute ~~said~~ a set of task procedures in ~~said~~ a sequential order according to said user program;

keep track of ~~said~~ a set of rollback procedures ~~in said transaction manager~~ corresponding to said set of task procedures, each of said set of procedures being determined based on a combination corresponding to an executed task procedure contained in said set of task procedures, said combination being contained in said plurality of combinations specified in said user program; and

execute said set of rollback procedures in a reverse order of said sequential order if said atomic transaction is to be aborted, wherein said rollback procedures are identified according to said keeping~~,~~

~~wherein said program logic is contained in a user program designed by a programmer~~.

Claim 17 (Original): The computer system of claim 16, wherein said transaction identifier is unique to each of the atomic transactions.

Claim 18 (Previously Presented): The computer system of claim 16, wherein said processing unit is operable to store data representing said rollback procedures in a stack to perform said keep.

Claim 19 (Original): The computer system of claim 18, wherein said stack is stored in a memory.

Claim 20 (Original): The computer system of claim 16, wherein said processing unit is further operable to examine a status returned by execution of one of said task procedures and to perform said aborting if said status indicates an error.

Claim 21 (Previously Presented): The computer system of claim 16, wherein said processing unit is operable to execute said rollback procedures asynchronously.

Claims 22 - 24 (Canceled)

Claim 25 (New): The computer readable medium of claim 7, wherein said rollback procedure is specified as a separate procedure from said task procedure in said user program.